WHAT IS CLAIMED IS

1. A computer which performs parallel processing of a plurality of programs in a time-division fashion, comprising:

hardware resources divided into a plurality of areas;

an evacuation unit which records identification information identifying a first program, and evacuates information stored in an area of said plurality of areas if the area is necessary for execution of a second program and is being used for execution of the first program; and

a restoration unit which restores the evacuated information to the area based on the identification information when the second program comes to a halt or to an end.

2. The computer as claimed in claim 1, further comprising an interruption unit which brings about interruption processing if the area is necessary for execution of a second program and is being used for execution of the first program, wherein said evacuation unit operates as part of the interruption processing to record the identification information and to evacuate the information stored in the area.

3. A computer which performs parallel processing of a plurality of programs in a time-division fashion, comprising:

hardware resources divided into a plurality of areas;

an evacuation unit which records identification information identifying a first program, and evacuates information stored in a first area of said plurality of areas if the first area and a second area of said plurality of areas are necessary for execution of a second program and are being used for execution of the first program, said evacuation unit subsequently evacuating information stored in the second area when use of the second area becomes actually necessary for execution of the second program; and

a restoration unit which restores the evacuated information to the first and second areas based on the identification information when the second program comes to a halt or to an end.

4. A method of controlling a computer which performs parallel processing of a plurality of programs in a time-division fashion, comprising the steps of:

providing hardware resources divided into a plurality of areas;

recording identification information identifying a first program, and evacuating information stored in an area of said plurality of areas if the area is necessary for execution of a second program and is being used for execution of the first program; and

restoring the evacuated information to the

area based on the identification information when the second program comes to a halt or to an end.

5. A method of controlling a computer which performs parallel processing of a plurality of programs in a time-division fashion, comprising the steps of:

providing hardware resources divided into a plurality of areas;

recording identification information identifying a first program, and evacuating information stored in a first area of said plurality of areas if the first area and a second area of said plurality of areas are necessary for execution of a second program and are being used for execution of the first program, followed by subsequently evacuating information stored in the second area when use of the second area becomes actually necessary for execution of the second program; and

restoring the evacuated information to the first and second areas based on the identification information when the second program comes to a halt or to an end.

6. A method of pipeline processing that attends to computation by connecting a central processing unit to an additional computation unit, comprising the steps of:

storing a computation instruction supplied to the computation unit;

executing the stored computation

instruction, and checking if completing the execution of the computation instruction requires more than a predetermined time length;

shifting the stored computation instruction to a dedicated storage if completing the execution of the computation instruction requires more than the predetermined time length; and

executing the computation instruction stored in the dedicated storage until the execution of the computation instruction is completed.

7. The method as claimed in claim 6, further comprising a step of successively outputting results of the execution of the computation instruction if the computation instruction is not an instruction requiring more than the predetermined time length in order to complete the execution.

8. An apparatus for pipeline processing in which a central processing unit is connected to an additional computation unit to attend to computation, comprising:

a first storage unit storing a computation instruction supplied to the computation unit;

a first computation unit which executes the computation instruction stored in said first storage unit;

a second storage unit which stores the computation instruction executed by the first computation unit if completing the execution of the computation instruction requires more than a

predetermined time length; and

a second computation unit which executes the computation instruction stored in the second storage unit until the execution of the computation instruction is completed.

9. An apparatus for pipeline processing in which a central processing unit is connected to an additional computation unit to attend to computation, comprising:

a first storage unit storing a computation instruction supplied to the computation unit;

a first computation unit which executes the computation instruction stored in said first storage unit;

second storage units, one of which stores the computation instruction executed by the first computation unit if completing the execution of the computation instruction requires more than a predetermined time length;

an indication unit which indicates an order of issuance of computation instructions stored in said second storage units; and

a second computation unit which executes a first-issued instruction among the computation instructions stored in said second storage units by selecting the first-issued instruction based on an indication of said indication unit until the execution of the first-issued instruction is completed.

10. An apparatus for pipeline processing in which a central processing unit is connected to a plurality of additional computation units to attend to computation, comprising:

a first storage unit which is provided in each of the computation units, and stores a computation instruction supplied to each of the computation units;

a first computation unit which is provided in each of the computation units, and executes the computation instruction stored in said first storage unit;

second storage units, each of which is provided in a corresponding one of the computation units, and stores the computation instruction executed by the first computation unit if completing the execution of the computation instruction requires more than a predetermined time length;

an indication unit which stores values indicative of an order of issuance of computation instructions stored in said second storage units; and

a second computation unit which executes a first-issued instruction among the computation instructions stored in said second storage units by selecting the first-issued instruction based on an indication of said indication unit until the execution of the first-issued instruction is completed, wherein an order of priority is determined in advance such that the values are stored in said indication unit in said order of priority.

11. The apparatus as claimed in claim 8, wherein a computation instruction requiring more than the predetermined time length for execution thereof is a multi-cycle computation instruction that requires a plurality of cycles before completion of execution thereof.

12. The apparatus as claimed in claim 9, wherein a computation instruction requiring more than the predetermined time length for execution thereof is a multi-cycle computation instruction that requires a plurality of cycles before completion of execution thereof.

13. The apparatus as claimed in claim 10, wherein a computation instruction requiring more than the predetermined time length for execution thereof is a multi-cycle computation instruction that requires a plurality of cycles before completion of execution thereof.

14. A divider, comprising:

a carry save adder; and

a full adder connected in series with said carry save adder, wherein the series connection of said carry save adder and said full adder performs an addition computation necessary for division computation.

15. The divider as claimed in claim 14, wherein said divider is a recursive-type divider.

16. The divider as claimed in claim 15, wherein the series connection of said carry save adder and said full adder obtains a sum of a portion of a dividend, a divider, and double the divider.

17. The divider as claimed in claim 16, wherein said divider is a recursive-type divider of a base number equal to four.